

Listing of Claims

1 Claim 1 (Previously Presented): A method of implementing an atomic transaction
2 using a program logic, said method comprising:

3 requesting in said program logic a transaction identifier for said atomic transaction;

4 generating said transaction identifier in a transaction manager in response to said
5 requesting;

6 specifying in said program logic a plurality of combinations for execution in a
7 sequential order, wherein each of said plurality of combinations contains said transaction
8 identifier, a task procedure, and a rollback procedure, wherein said task procedure
9 implements a part of said atomic transaction and said rollback procedure is designed to
10 rollback said task procedure;

11 executing said task procedures in said sequential order;

12 keeping track of said rollback procedures in said transaction manager; and

13 executing said rollback procedures in a reverse order of said sequential order if said
14 atomic transaction is to be aborted, wherein said rollback procedures are identified according
15 to said keeping.

1 Claim 2 (Original): The method of claim 1, wherein said transaction identifier is
2 unique to each of the atomic transactions.

1 Claim 3 (Previously Presented): The method of claim 1, wherein said keeping
2 comprises storing data representing said rollback procedures in a stack.

1 Claim 4 (Original): The method of claim 3, wherein said stack is stored in a memory.

1 Claim 5 (Original): The method of claim 1, further comprising examining a status
2 returned by execution of one of said task procedures and performing said aborting if said
3 status indicates an error.

1 Claim 6 (Original): The method of claim 1, wherein said aborting is performed
2 asynchronously.

1 Claims 7 (Previously Presented): A computer readable medium carrying one or more
2 sequences of instructions representing a program logic for execution on a system, said
3 program logic implementing an atomic transaction, wherein execution of said one or more
4 sequences of instructions by one or more processors contained in said system causes said one
5 or more processors to perform the actions of:

6 requesting an identifier for said atomic transaction;

7 setting a variable to equal said identifier;

8 specifying a plurality of combinations for execution, wherein each of said plurality of
9 combinations contains said transaction identifier, a task procedure, and a rollback procedure,
10 wherein said task procedure implements a part of said atomic transaction and said rollback
11 procedure is designed to rollback said task procedure; and

12 aborting said atomic transaction by specifying said identifier associated with an abort
13 procedure to cause said rollback procedures to be executed.

1 Claim 8 (Original): The computer readable medium of claim 7, wherein said
2 specifying comprises including each of said plurality of combinations in a single procedure
3 call.

1 Claim 9 (Original): The computer readable medium of claim 7, further comprising
2 examining a status returned by execution of one of said task procedures and performing said
3 aborting if said status indicates an error.

1 Claims 10 - 15 (Canceled)

1 Claim 16 (Previously Presented): A computer system comprising: a memory storing
2 a plurality of instructions; and a processing unit coupled to said memory and executing said
3 plurality of instructions to support implementation of an atomic transaction in a programming
4 environment, said processing unit being operable to:

5 request in a program logic a transaction identifier for said atomic transaction;

6 generate said transaction identifier in a transaction manager in response to said

7 requesting;

8 specify in said program logic a plurality of combinations for execution in a sequential
9 order, wherein each of said plurality of combinations contains said transaction identifier, a
10 task procedure, and a rollback procedure, wherein said task procedure implements a part of
11 said atomic transaction and said rollback procedure is designed to rollback said task
12 procedure;

13 execute said task procedures in said sequential order;

14 keep track of said rollback procedures in said transaction manager; and

15 execute said rollback procedures in a reverse order of said sequential order if said
16 atomic transaction is to be aborted, wherein said rollback procedures are identified according
17 to said keeping.

1 Claim 17 (Original): The computer system of claim 16, wherein said transaction
2 identifier is unique to each of the atomic transactions.

1 Claim 18 (Previously Presented): The computer system of claim 16, wherein said
2 processing unit is operable to store data representing said rollback procedures in a stack to
3 perform said keep.

1 Claim 19 (Original): The computer system of claim 18, wherein said stack is stored
2 in a memory.

1 Claim 20 (Original): The computer system of claim 16, wherein said processing unit
2 is further operable to examine a status returned by execution of one of said task procedures
3 and to perform said aborting if said status indicates an error.

1 Claim 21 (Previously Presented): The computer system of claim 16, wherein said
2 processing unit is operable to execute said rollback procedures asynchronously.